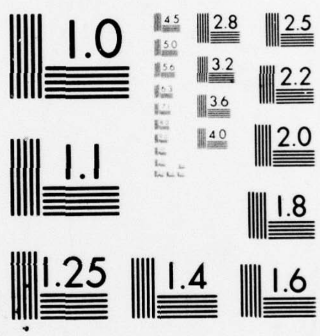
AD-A048 459 MASSACHUSETTS INST OF TECH CAMBRIDGE RESEARCH LAB OF--ETC F/G 9/4
A UNIFIED FRAMEWORK FOR THE REALIZATION PROBLEM IN LINEAR DIGIT--ETC(U)
NOV 77 D S CHAN DAAB07-76-C-1400

UNCLASSIFIED NL

1 OF 1
ADA048459

END
DATE
FILMED
2-78
DDC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

12

AD A048459

AD No.

DDC FILE COPY

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A UNIFIED FRAMEWORK FOR THE REALIZATION PROBLEM IN LINEAR DIGITAL FILTERING		5. TYPE OF REPORT & PERIOD COVERED Reprint
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) David S. K. Chan		8. CONTRACT OR GRANT NUMBER(s) DAAB07-76-C-1400
9. PERFORMING ORGANIZATION NAME AND ADDRESS Research Laboratory of Electronics Massachusetts Institute of Technology Cambridge, Massachusetts 02139		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 12 6p.
11. CONTROLLING OFFICE NAME AND ADDRESS U. S. Army Research Office P. O. Box 12211 Research Triangle Park North Carolina 27709		12. REPORT DATE November 1977
		13. NUMBER OF PAGES 6
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

DDC
JAN 5 1978
F

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Reprinted from Proc. 1977 Int. Conf. on Acoustics, Speech, and Signal Processing pp. 543-547 (1977)

w/o location

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Linear filters
Digital filters
Digital signal processing

+ are treated

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

In a new approach to characterizing the implementation of linear digital filters ~~we treat~~ all possible realizations for any filter in the same framework, based on the representation of an input/output specification for a linear filter by a unique, possibly infinite transmission matrix, which can be extended to include arrays of higher than two dimensions or of arbitrary dimensions. A set of notations and operation rules for these arrays and matrices is presented, and all possible implementations of a filter are characterized as factorizations of its transmission array.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE 304050 UNCLASSIFIED

A UNIFIED FRAMEWORK FOR THE REALIZATION PROBLEM IN LINEAR DIGITAL FILTERING

David S. K. Chan

Massachusetts Institute of Technology
Research Laboratory of Electronics
Cambridge, Mass. 02139

Abstract

A new approach to characterizing the problem of implementing linear digital filtering operations on finite-state machines is presented. In this formulation, the implementation of shift-invariant or shift-varying, one-dimensional or multi-dimensional filters as single-input or multiple-input, time-invariant or time-varying systems are all treated in the same framework. This framework is based on the representation of an input/output specification for a linear filter by a unique, possibly infinite transmission matrix. To treat two- or higher dimensional filters, the concept of transmission matrix is extended to include arrays of higher than two dimensions. The framework also uses matrices whose entries are arrays of arbitrary dimensions. A set of notations and operation rules for these arrays and matrices is presented. Within this framework, all possible implementations of a filter are characterized as factorizations of its transmission array.

I. INTRODUCTION

The problem of finding implementations for linear digital filters has been approached in many ways. For a given realizable filter, there exist various alternative implementations. However, there has been no common framework in which all possible realizations for any filter can be described. For example, though it is possible to describe all time-invariant, single-input-single-output realization structures for 1-D shift-invariant filters by a signal flow graph [1], such a graph does not extend naturally to include many alternative implementations which use multiple or block inputs [2,3]. Moreover, the signal flow graph is only capable of describing time-invariant realizations of shift-invariant filters, and does not extend naturally to shift-varying filters.

The signal flow graph has been extended to characterize 2-D filter structures by including two types of z^{-1} branches [4]. However, as a characterization of implementations, this type of flowgraph is even less attractive than the 1-D version, since it fails to characterize how data actually flows through an implementation, and its z^{-1} branches do not correspond to storage cells.

This work was supported by the Joint Services Electronics Program under contract no. DAAB07-76-C-1400

The actual amount of storage required depends on the order in which data is processed, which is not specified by this representation. The problem with trying to characterize implementations of multi-dimensional filters by a generalization of this type is that, though the input and output data structure of say, a 2-D filter, is different from that of a 1-D filter, there is actually no difference between the inherent structure of a machine which implements 1-D filters and one which implements 2-D filters. Thus, rather than having two different descriptions for 1-D and 2-D filter implementations, one common framework should suffice for all filters, regardless of dimension.

In this paper, a framework is presented which not only characterizes all possible realizations (single or multiple input, time-invariant or varying) of any linear filter, but also treats shift-invariant and shift-varying filters of any dimension in the same way. Within this framework, a definitive answer is given to the question; what constitutes a realization of a multi-dimensional digital filter? Much of this framework for the case of single-input-single-output implementations has been presented in [5] in a less general form.

II. DEFINITIONS AND NOTATIONS

A digital filter may be defined to be a rule for transforming digital signals within a given class. A digital signal may be any countable set of numbers. However, usually it is representable as an indexed sequence of numbers, where the number of indices is called the dimension of the signal. The dimension of the signals a filter transforms is in turn called the dimension of the filter.

If the numbers in a digital signal were regarded to have continuous and unbounded magnitudes, then digital signals of the same type may be regarded as elements of a linear function space with a countable basis. A digital filter transforming such signals may then be described as an operator on the space. In particular, a linear filter, corresponding to a linear operator, would be uniquely describable in input/output behavior by a possibly infinite array of numbers, which specify the transformation of basis elements. This array will be called the transmission array of a linear

filter.

A digital signal can also be described as an array of numbers. The number of indices associated with each entry of the array corresponds to the dimension of the signal, and is called the dimension of the array. An array will be denoted by a symbol having superscripts and/or subscripts, which correspond to the indices associated with each array entry. These scripts may be any letters, with or without numerical subscripts, provided they are distinct and occur, from left to right, in alphabetical order and order of ascending numerical subscripts among similar letters. The size of an array is given by the numbers of numerical values taken on by the indices. As a convention, subscript indices, from left to right, are mentioned before superscript indices, also from left to right. For example, the size of the array a_j^{st}, where $0 \le s < 2$, $0 \le t < 3$, $0 \le j < 5$, is 5x2x3.

A slice of an array is defined to be a new array formed by retaining only those entries where a subset of the indices take on certain specified values. To denote a slice of an array, the indices given specified values are either replaced by these values if they are numbers, or by them enclosed in angular brackets if they are symbols. For example, $a^i_{<mn>}$ is a 1-D slice of the 3-D array a^i_{st} consisting of those entries of a^i_{st} where s,t take on the values m,n respectively. Also, $b^{<i>}_{3,4}$ denotes the entry of b^s_{mn} with values 3,4,i for its indices m,n,s respectively.

If two or more arrays have the same superscripts and subscripts, then their sum is defined to be an array with these same superscripts and subscripts, and with entries equal to the sum of corresponding entries. If two or more arrays do not share common superscripts or common subscripts, their tensor product is defined. This product is said to be uncontracted if no symbol appears both as superscript in one factor and subscript in another, and contracted otherwise. If a symbol is thus repeated, summation over all possible values of that index is implied. For example,

$$a^{ij}_{mn}\, b^{n}_{ip} = \sum_{s} \sum_{t} a^{<s>j}_{m<t>}\, b^{<t>}_{<s>p} \tag{1}$$

This is the well-known summation convention due to Einstein. Each summation over a repeated index is called a contraction.

The tensor product takes on, respectively as superscripts and subscripts, the combined superscripts and subscripts of its factors, less any repeated symbols, each set being arranged in alphabetical and numerical order. Thus an uncontracted product contains, as entries, all possible products made up of one entry from each factor array, and has a dimension equal to the sum of the dimensions of its factors. A contracted product is similar to an uncontracted product, except that its dimension is two lower for each repeated index, because of contraction.

It should be clear that sum and tensor product of arrays are commutative and associative operations. The latter is also distributive over the former. We shall define a non-commutative product as follows. If the number of superscripts in one array is equal to the number of subscripts in another array, and these indices take on corresponding ranges of values, then the chain product of the first array with the second, in that order, is defined to be the contracted tensor product of the arrays, formed by setting the superscripts of the first array equal to the subscripts of the second, and keeping the other indices distinct.

In terms of the notations and definitions developed thus far, a 1-D linear digital filter is described in input/output behavior by

$$y_m = \Omega^i_m\, x_i \tag{2}$$

where Ω^i_m is the transmission array of the filter, and x_i, y_m are the input and output signal arrays respectively. Similarly, a 2-D filter is described by

$$y_{mn} = \Omega^{ij}_{mn}\, x_{ij} \tag{3}$$

Thus, with the convention that signals be represented by arrays with subscripts only, and transmission arrays by arrays with equal numbers of subscripts and superscripts, the output of a filter is given by the chain product of its transmission array with its input array.

Before ending this section, one more mathematical entity will be defined, viz. a generalized matrix. It is defined to be a matrix whose entries are arbitrary dimensional arrays, such that those in the same row (column) have the same subscripts (superscripts). Ordinary and block matrices are special cases of generalized matrices. The sum of two generalized matrices is defined in the obvious manner if corresponding entries in the matrices have the same superscripts and subscripts. The product of two generalized matrices is defined when the first matrix has the same number of columns as the number of rows in the second matrix, and the number of superscripts and their ranges of values in each entry in the i^{th} column of the first matrix are respectively equal to the number of subscripts and their ranges in the i^{th} row of the second matrix, for all i. When defined, this product is computed just like an ordinary matrix product, only that the products of individual entries are computed as chain products of the arrays.

It is convenient to have an abbreviated common notation for arrays and generalized matrices. Thus, we shall reserve underlined, possibly subscripted capital letters to represent them. When two such symbols are placed side by side, either the chain product or generalized matrix product is implied. Furthermore, the symbol "$\underline{O}$" will be reserved to denote a zero array or zero generalized matrix, i.e., an entity with all entries equal to zero, and "$\underline{I}$" reserved to denote a unit array

or unit generalized matrix, defined as follows. A unit array is an array of the form $I^{s_1 \ldots s_N}_{r_1 \ldots r_N}$, where the range of s_i coincides with the range of r_i for all i, and $I^{<s_1 \ldots s_N>}_{<r_1 \ldots r_N>}$ equals one if $<s_1 \ldots s_N> = <r_1 \ldots r_N>$ and zero otherwise. A unit generalized matrix is characterized by unit arrays along its diagonal and zero arrays elsewhere.

III. THE NEW FRAMEWORK

The realization problem for linear digital filters may be stated as follows: Given the input/output description of a filter, find an algorithm which realizes this filter on a finite-state machine to some given degree of accuracy. The objective of this section is to characterize solutions to this problem. We shall focus attention on realization by quasi-linear machines, defined as follows.

<u>Definition 1</u>: A <u>quasi-linear machine</u> is a machine whose operation can be described by the equations

$$\begin{aligned} \underline{S}_{k+1} &= \underline{A}_k \odot \underline{S}_k \oplus \underline{B}_k \odot \underline{X}_k \\ \underline{Y}_k &= \underline{C}_k \odot \underline{S}_k \oplus \underline{D}_k \odot \underline{X}_k \end{aligned} \qquad k = 1,2,\ldots \tag{4}$$

where $\underline{A}_k$, $\underline{B}_k$, $\underline{C}_k$, $\underline{D}_k$, $\underline{S}_k$, $\underline{X}_k$, and $\underline{Y}_k$, k=1,2,..., are arrays of arbitrary though compatible and bounded dimensions and sizes, whose elements are numbers representable by a finite number, say L, of digits each (usually binary digits, or bits). The symbols $\odot$ and $\oplus$ denote L-digit-precision products and sums using rounding. The number L is called the <u>wordlength</u> of the machine.

In (4), the subscript k may be thought of as a time marker for the machine, which operates as follows. At k=1 the first operation begins. Then, at each subsequent integer value of k, an operation completes while a new one is begun. $\underline{X}_k$ represents a set of numbers fed into the machine at time k, $\underline{Y}_k$ represents a set of numbers retrieved from the machine at time k+1, and $\underline{S}_k$ represents a set of numbers stored in the machine at time k. Realizability of a linear digital filter may now be defined as follows.

<u>Definition 2:</u> A linear digital filter is <u>realizable</u> if there exists a quasi-linear machine and an order of enumerating the elements of the filter's input array, such that given any $\varepsilon>0$ and $\delta>0$, there is a wordlength L with which the machine can compute any element of the filter's output array, in finite time and to within an accuracy tolerance of ε, if the elements of the input array are bounded in magnitude by δ, and are fed into the machine in the order mentioned above.

The parameters ε and δ in the above definition together determine the dynamic range required of a realization. Implicit in the definition is that a realizable filter must be bounded-input-bounded-output (BIBO) stable, since the output array elements must be bounded to be representable by a finite number of digits, and if the filter were not stable, every error tolerance would be exceeded by some output point. The following theorem characterizes the realizability of a filter in terms of its transmission array, and is stated without proof because of space.

<u>Theorem 1</u>: A linear digital filter is realizable if and only if it is BIBO stable and its transmission array, $\Omega^{i_1 \ldots i_N}_{j_1 \ldots j_N}$, can be factorized into the form

$$\Omega^{i_1 \ldots i_N}_{j_1 \ldots j_N} = \underline{P} \left(\prod_{k=1}^{\infty} \underline{\Psi}_k \right) \underline{Q} \tag{5}$$

where

$$\prod_{k=1}^{\infty} \underline{\Psi}_k = \ldots \underline{\Psi}_2 \underline{\Psi}_1 \tag{6}$$

$$\underline{\Psi}_k = \begin{bmatrix} \underline{A}_k & \underline{0} & \ldots & \underline{0} & \underline{B}_k & \underline{0} & \ldots \\ \underline{0} & \underline{I} & \underline{0} & \ldots & \underline{0} & . & \\ . & . & . & . & . & & \\ . & \underline{0} & . & . & . & . & \\ . & . & . & . & .. & & \\ \underline{0} & . & . & \underline{I} & \underline{0} & . & \\ \underline{C}_k & \underline{0} & \ldots & .\underline{0} & \underline{D}_k & \underline{0} & \ldots \\ \underline{0} & & \ldots & & \underline{0} & \underline{I} & \\ . & & & & . & & . \\ . & & & & . & & \quad . \\ . & & & & . & & \qquad . \end{bmatrix} \begin{matrix} \\ \\ \\ \\ \\ \\ \leftarrow (k+1)^{th} \text{ row} \\ \\ \\ \\ \\ \end{matrix} \tag{7}$$

$\underline{A}_k$, $\underline{B}_k$, $\underline{C}_k$, $\underline{D}_k$ are arrays of arbitrary though compatible and bounded dimensions and sizes, with entries bounded in magnitude,

$$\underline{P} = [\underline{0}\ \underline{F}_1\ \underline{F}_2\ \ldots] \qquad \underline{Q} = \begin{bmatrix} \underline{0} \\ \underline{G}_1 \\ \underline{G}_2 \\ \vdots \end{bmatrix} \tag{8}$$

and $\underline{F}_k$, $\underline{G}_k$ are arrays satisfying the following: 1) The entries of $\underline{F}_k$ ($\underline{G}_k$) for all k corresponding to each set of numerical subscripts (superscripts) are all zero except for one entry in one $\underline{F}_k$ ($\underline{G}_k$), which equals one; 2) For each k, all but at most one entry of $\underline{F}_k$ ($\underline{G}_k$) corresponding to each set of numerical superscripts (subscripts) are zero.

The interpretation of the generalized matrices and arrays in theorem 1 is as follows. The matrix $\underline{Q}$ represents a mapping of the possibly infinite filter input array, say $\underline{X}$, into a sequence of finite arrays. If this sequence of arrays is denoted $\underline{X}_1$, $\underline{X}_2$,..., and if we let $\underline{Q}\ \underline{X}$ =

$(\underline{S}_1, \underline{X}_1, \underline{X}_2, \ldots)$, where $\underline{S}_1 = \underline{0}$, then the action of each $\underline{\Psi}_k$ may be regarded as converting $\underline{S}_k$ to $\underline{S}_{k+1}$ and $\underline{X}_k$ to $\underline{Y}_k$, where $\underline{S}_{k+1}$ and $\underline{Y}_k$ satisfy

$$\begin{aligned} \underline{S}_{k+1} &= \underline{A}_k \underline{S}_k + \underline{B}_k \underline{X}_k \\ & \qquad k = 1,2,\ldots \qquad \underline{S}_1 = \underline{0} \\ \underline{Y}_k &= \underline{C}_k \underline{S}_k + \underline{D}_k \underline{X}_k \end{aligned} \tag{9}$$

The rest of the inputs to $\underline{\Psi}_k$ are left unchanged by it, thus $\underline{\Psi}_k \ldots \underline{\Psi}_1 \underline{Q}\ \underline{X} = (\underline{S}_{k+1}, \underline{Y}_1, \ldots, \underline{Y}_k, \underline{X}_{k+1}, \ldots)$, $k \geq 1$. Now if in (9), $\underline{A}_k, \underline{B}_k, \underline{C}_k, \underline{D}_k$, and $\underline{X}_k$ are all bounded, and the filter is BIBO stable, it can be shown that the computation of (9) can be performed, for all k and to any desired accuracy, by a quasi-linear machine with a sufficiently long wordlength. Thus, using this machine, any desired element of the filter output array can be found in the array $\underline{P}(\underline{\Psi}_k \ldots \underline{\Psi}_1)\underline{Q}\ \underline{X}$ for sufficiently large k. The matrix $\underline{P}$ represents a mapping of the outputs of the quasi-linear machine into the possibly infinite filter output array.

Because of theorem 1, every realization of a linear digital filter can be characterized by the following:

Definition 3: A realization for a linear digital filter with transmission array $\Omega^{i_1 \ldots i_N}_{j_1 \ldots j_N}$ is a set of generalized matrices, $[\underline{P}, \underline{\Psi}_1, \underline{\Psi}_2, \ldots, \underline{Q}]$, satisfying the conditions in theorem 1, such that

$$\Omega^{i_1 \ldots i_N}_{j_1 \ldots j_N} = \underline{P} \left(\prod_{k=1}^{\infty} \underline{\Psi}_k \right) \underline{Q}\ . \tag{10}$$

$\underline{P}$ and $\underline{Q}$ are called the I/O matrices and $\underline{\Psi}_k$, the transmission factors, of the realization.

Definition 4: A realization is said to be regular if the dimensions and sizes of the arrays $\underline{A}_k, \underline{B}_k, \underline{C}_k, \underline{D}_k$ in $\underline{\Psi}_k$, $k \geq 1$, are independent of k.

Theorem 2: To every realizable linear digital filter there corresponds a regular realization.

Regular realizations are interesting because real machines generally have fixed-size input/output channels and storage capacities. Since the important information in each transmission factor $\underline{\Psi}_k$ is contained in the finite arrays $\underline{A}_k, \underline{B}_k, \underline{C}_k, \underline{D}_k$, it is convenient to specify realizations in terms of these arrays only, rather than the entire transmission factors. Thus we shall define the matrix $\begin{bmatrix} \underline{A}_k & \underline{B}_k \\ \underline{C}_k & \underline{D}_k \end{bmatrix}$ to be the reduced transmission factor corresponding to $\underline{\Psi}_k$, and specify the realization in definition 3 alternatively as a set $[\underline{P}, \underline{\Phi}_1, \underline{\Phi}_2, \ldots, \underline{Q}]$, where $\underline{\Phi}_k$ is the reduced transmission factor corresponding to $\underline{\Psi}_k$, k=1,2,... Unless otherwise stated, all realizations in the following will be in this form.

A realization as defined represents a scheme for implementing a filter by segmenting the input data into finite portions and operating on one portion at a time. Given such a scheme, there remains infinitely many ways in which the processing of each finite portion of data can be accomplished. In particular, any factorization of $\underline{\Phi}_k$ represents a way of performing, by a sequence of operations, the processing by the k^{th} transmission factor, namely

$$\begin{bmatrix} \underline{S}_{k+1} \\ \underline{Y}_k \end{bmatrix} = \begin{bmatrix} \underline{A}_k & \underline{B}_k \\ \underline{C}_k & \underline{D}_k \end{bmatrix} \begin{bmatrix} \underline{S}_k \\ \underline{X}_k \end{bmatrix} = \underline{\Phi}_k \begin{bmatrix} \underline{S}_k \\ \underline{X}_k \end{bmatrix} \tag{11}$$

Thus, a more detailed description of a realization is the following:

Definition 5: A realization structure for a filter with transmission array $\Omega^{i_1 \ldots i_N}_{j_1 \ldots j_N}$ is a set of generalized matrices, $[\underline{P}, \{\underline{\Lambda}_{1i}, \ldots, \underline{\Lambda}_{L_i i}\}_{i=1}^{\infty}, \underline{Q}]$, where $\underline{\Lambda}_{ji}$ is of bounded size and $\underline{\Phi}_i \overset{\Delta}{=} \underline{\Lambda}_{L_i i} \ldots \underline{\Lambda}_{1i}$ is well defined, such that $[\underline{P}, \underline{\Phi}_1, \underline{\Phi}_2, \ldots \underline{Q}]$ is a realization for the filter.

A useful classification of realizations and structures follows.

Definition 6: A regular realization $[\underline{P}, \underline{\Phi}_1, \underline{\Phi}_2, \ldots, \underline{Q}]$ is time-invariant if $\underline{\Phi}_k$ is independent of k, in which case it may be written as $[\underline{P}, \underline{\Phi}, \underline{Q}]$. A time-invariant realization structure is specified in the form $[\underline{P}, \underline{\Lambda}_1, \ldots, \underline{\Lambda}_{N_p}, \underline{Q}]$.

Clearly, given a realizable filter, there exists many possible choices for the I/O matrices $\underline{P}$ and $\underline{Q}$. Not all of these lead to practical realizations. Given two I/O matrices which support a regular realization, all possible regular realizations which use these I/O matrices are related simply as follows.

Theorem 3: $[\underline{P}, \underline{\Phi}_1, \underline{\Phi}_2, \ldots, \underline{Q}]$ and $[\underline{P}, \hat{\underline{\Phi}}_1, \hat{\underline{\Phi}}_2, \ldots, \underline{Q}]$ are regular realizations for the same filter if and only if there exist invertible arrays $\underline{T}_k$, k=1,2,..., such that

$$\hat{\underline{\Phi}}_k = \begin{bmatrix} \underline{T}_{k+1} & \underline{0} \\ \underline{0} & \underline{I} \end{bmatrix} \underline{\Phi}_k \begin{bmatrix} \underline{T}_k^{-1} & \underline{0} \\ \underline{0} & \underline{I} \end{bmatrix} \qquad k=1,2,\ldots \tag{12}$$

The special case of time-invariant realization structures is worth mentioning as a corollary to theorem 3.

Corollary 1: Two time-invariant realization structures, $[\underline{P},\underline{\Gamma}_1,\ldots,\underline{\Gamma}_{N_1},\underline{Q}]$ and $[\underline{P},\underline{\Lambda}_1,\ldots,\underline{\Lambda}_{N_2},\underline{Q}]$, realize the same filter if and only if an invertible array $\underline{T}$ exists such that

$$\underline{\Gamma}_{N_1}\cdots\underline{\Gamma}_1 = \begin{bmatrix} \underline{T} & \underline{0} \\ \underline{0} & \underline{I} \end{bmatrix} \underline{\Lambda}_{N_2}\cdots\underline{\Lambda}_1 \begin{bmatrix} \underline{T}^{-1} & \underline{0} \\ \underline{0} & \underline{I} \end{bmatrix} \tag{13}$$

Time-invariant single-input-single-output structures for shift-invariant filters are the type of realization structures that have been most often studied. A filter with transmission array $\Omega^{i_1\ldots i_N}_{j_1\ldots j_N}$ is said to be shift-invariant if $\Omega^{<i_1\ldots i_N>}_{<j_1\ldots j_N>}$ depends on i_k and j_k only through $i_k-j_k, k=1,\ldots,N$. That is,

$$\Omega^{<i_1\ldots i_N>}_{<j_1\ldots j_N>} = \Omega^{<m_1\ldots m_N>}_{<n_1\ldots n_N>} \text{ whenever } i_k-j_k=m_k-n_k, \quad 1\le k\le N \tag{14}$$

The slice $\Omega^{0\ldots 0}_{j_1\ldots j_N}$ of such a filter's transmission array is called its impulse response. A shift-invariant filter is classified as FIR if its impulse response has a *finite number of nonzero* entries, and IIR otherwise. The following theorems consider the realizability of shift-invariant 1-D and 2-D filters.

Theorem 4: A 1-D FIR filter is always realizable. A 1-D IIR filter with transmission array Ω^i_j is realizable if and only if $\sum_{j=0}^{\infty} \Omega^0_{<j>} z^j$ is a rational function of z.

Definition 7: Let the impulse response samples of a 2-D shift-invariant filter be arranged equally spaced on a planar grid. If all the nonzero samples lie on a straight line, then the filter is said to be degenerate.

Theorem 5: A nondegenerate 2-D FIR filter is realizable if and only if one of the subscripts of its transmission array has finite range. A nondegenerate 2-D IIR filter is not realizable unless one of the subscripts of its transmission array has finite range.

IV. AN EXAMPLE

Several examples of single-input-single-output realization structures described in this framework have been given in [5]. An example of a time-invariant realization which processes 4 input data samples at a time is given here. Let I_k denote a k×k identity matrix and O_k a k×k zero matrix. Then $[Q^T,\Phi,Q]$ is such a realization for the 1-D FIR filter with transmission array Ω if

$$\Omega = \begin{bmatrix} h_0 & & & \\ h_1 & h_0 & & \\ h_2 & h_1 & & \\ & h_2 & \ddots & \end{bmatrix}$$

$$Q = \begin{bmatrix} O_2 & O_2 & O_2 & O_2 & \cdots \\ \hline I_4 & O_4 & O_4 & \cdots & \\ \hline O_4 & I_4 & O_4 & \cdots & \\ \hline \vdots & & & & \end{bmatrix}$$

$$\Phi = \left[\begin{array}{cc|cccc} & & & & 1 & \\ & & & & & 1 \\ \hline h_2 & h_1 & h_0 & & & \\ & h_2 & h_1 & h_0 & & \\ & & h_2 & h_1 & h_0 & \\ & & & h_2 & h_1 & h_0 \end{array}\right]$$

Furthermore, a realization structure can be constructed by noting that Φ can be factorized as

$$\Phi = \left[\begin{array}{c|c} I_2 & \\ \hline O_2 & \\ & I_4 \\ O_2 & \end{array}\right] \left[\begin{array}{c|c} I_2 & \\ \hline & H \end{array}\right] \left[\begin{array}{ccc} O_2 & O_2 & I_2 \\ \hline & I_6 & \end{array}\right]$$

where H defines a cyclic convolution and can be further factorized.

REFERENCES

[1] R. E. Crochiere and A. V. Oppenheim, "Analysis of Linear Digital Networks," Proc. IEEE, vol. 63, no. 4, pp. 581-595, April 1975.

[2] B. Gold and C. M. Rader, Digital Processing of Signals, New York: McGraw-Hill, 1969.

[3] C. S. Burrus, "Block Realization of Digital Filters," IEEE Trans. Audio Electroacoust., vol. AU-20, no. 4, pp. 230-235, Oct. 1972.

[4] S. K. Mitra, A. D. Sagar and N. A. Pendergrass, "Realizations of Two-Dimensional Recursive Digital Filters," IEEE Trans. Circuits Systems, vol. CAS-22, no. 3, pp. 177-184, March 1975.

[5] D.S.K. Chan, "A Novel Framework for the Description of Realization Structures for 1-D and 2-D Digital Filters," Record of the IEEE Electronics and Aerospace Systems Convention, Washington, DC, pp. 157A-157H, Sept. 1976.